

**AMENDMENTS TO THE CLAIMS**

The listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Previously Presented) A computer implemented method for adjusting a priority of an execution thread, said method comprising:
  - indicating that the execution thread needs a higher priority by updating a user mode accessible data area, the indicating performed without increasing a priority corresponding to the execution thread and the indicating including setting a critical section flag within the user mode accessible data;
  - indicating that the execution thread is entering a critical section of code;
  - updating a priority offset amount prior to the execution thread entering the critical section of code, whereupon the priority offset amount is included in the user mode accessible data;
  - during execution of the critical section of code:
    - detecting a preemption event, wherein the preemption event has a preemption event priority;
    - reading the user mode accessible data area in response to the detected preemption event;
    - determining whether a priority applied flag and the critical section flag have been set;
    - in response to detecting that the critical section flag has been set and the priority applied flag has not been set, raising the execution thread's priority by the priority offset amount and setting the priority applied flag;
  - after raising the execution thread's priority:
    - in response to the preemption event priority being greater than the execution thread's priority, preempting the execution thread;
    - and

in response to the preemption event priority not being greater than the execution thread's priority, allowing the execution thread to continue executing.

2. (Canceled)
3. (Canceled)
4. (Canceled)
5. (Previously Presented) The method of claim 1 further comprising:  
resetting the critical section flag within the user mode accessible data indicating that the execution thread is no longer in the critical section;  
receiving a second preemption event;  
determining that the critical section flag is no longer set and that the priority applied flag is set;  
lowering the execution thread's priority by the priority offset amount in response to the determination that the critical section flag is no longer set and that the priority flag is set; and  
resetting the priority applied flag indicating that the execution thread's priority is no longer raised.
6. (Canceled)
7. (Previously Presented) The method of claim 1 wherein the indicating that the execution thread needs a higher priority is performed in response to the execution thread entering the critical code section, and wherein the critical code section utilizes a shared system resource.
8. (Canceled)
9. (Canceled)
10. (Canceled)

11. (Canceled)

12. (Canceled)

13. (Canceled)

14. (Canceled)

15. (Canceled)

16. (Canceled)

17. (Canceled)

18. (Canceled)

19. (Canceled)

20. (Canceled)